1. gdb--命令
2. 前提条件 ，可执行文件必须包含调试信息 gcc -g
3. Gdb 文件名 -- 启动gdb调试
4. 查看代码的命令:

当前文件：

List 行号(函数名)

指定文件:

List 文件名:行号(函数名)

1. 设置断点

当前文件：

B 行号(函数名)

指定文件:

B 文件名:行号(函数名)

设置条件断点:

B 行号 if value==23

查看断点信息:

Info b

删除断点:

D 断点的编号

1. 开始调试

只执行一行代码:start

继续执行停在断点处:continue --c

直接停在断点处:run --r

1. 单步调试

进入函数体： step -- s

跳出函数体: finish 如果在循环处有断点，需要将断点删除

不进入函数体: next --n

1. 追踪函数

自动打印变量的值:display 变量名称

取消变量的追踪:undisplay 编号

获取编号: info display

手动打印变量的值: print --p

获取变量对应的类型: ptype

1. 跳出循环: u
2. 退出gdb :quit

Makefile

一个规则

三要素：目标，依赖，命令

目标：依赖

命令

1 第一条规则是用来生成终极目标的规则

1. 如果规则中的依赖不存在，向下寻找其它的规则
2. 更新机制：比较的是目标文件和依赖文件的时间

两个函数

1 查找指定的目录下，指定类型的文件

Src = $(wildcard “/aa/\*.c”)

2 匹配替换函数

Obj = $(patsubst %.c,%.o,$(src))

三个自动变量

1 $< 规则中的第一个依赖

2 $\* 规则中的所有依赖

3 $@ 规则中的目标

模式规则

%.c,%.o

Gcc -c $< -o $@

liunx系统IO函数